

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

JUL 29 1981

AgRISTARS

SR-L1-00306
JSC-17037

NASA-CR-161056

A Joint Program for
Agriculture and
Resources Inventory
Surveys Through
Aerospace
Remote Sensing

Supporting Research

June 1981

"AS-BUILT" DESIGN SPECIFICATION FOR SEGMENT
MAP (SGMAP) PROGRAM

E82-10094

CR-161056

M. A. Tompkins

(E82-10094) AS-BUILT DESIGN SPECIFICATION
FOR SEGMENT MAP (SGMAP) PROGRAM (Lockheed
Engineering and Management) 100 p
HC A05/MF A01

N82-21646

CSCI 02C

Unclass

G3/43 00094

Lockheed Engineering and Management Services Company, Inc.
1830 NASA Road 1, Houston, Texas 77058



Lyndon B. Johnson Space Center
Houston, Texas 77058

SR-L1-00306
JSC-17037

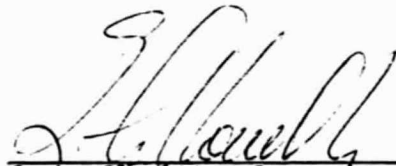
"AS-BUILT" DESIGN SPECIFICATION
FOR
SEGMENT MAP (SGMAP) PROGRAM

Job Order 71-308

Prepared By

M. A. Tompkins

Approved By



G. L. Clouette, Supervisor
Support Systems Software Section



R. Kent Lennington, Supervisor
Techniques Development Section



R. A. McClane, Manager
Data Systems Department



T. C. Minter, Manager
Development and Evaluation Department

Prepared By

Lockheed Engineering and Management Services Company, Inc.

For

Earth Observations Division
Space and Life Sciences Directorate

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

June 1981

LEMSCO-15937

PRECEDING PAGE BLANK NOT FILMED

1. Report No. JSC-17037, SR-L1-00306		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle "As-Built" Design Specification for Segment Map (SGMAP) Program				5. Report Date June 1981	
				6. Performing Organization Code SG2	
7. Author(s) Mary Ann Tompkins				8. Performing Organization Report No. LEMSCO-15937	
9. Performing Organization Name and Address Lockheed Engineering and Management Services Company, Inc., Systems and Services Division Houston, Texas 77058				10. Work Unit No.	
				11. Contract or Grant No. NAS 9-15800	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Lyndon B. Johnson Space Center Houston, Texas 77058 <i>Dr. G. Badhwar/SG3</i>				13. Type of Report and Period Covered "As-Built"	
				14. Sponsoring Agency Code	
15. Supplementary Notes					
16. Abstract <p>This document is the "As-Built" Design Specification for Segment Map (SGMAP) Program which is part of the CLASFYI package.</p> <p>This program is designed to output symbolic maps or numerical dumps from Cluster/Classification files or ground truth/processed ground truth files which are in 'Universal' format.</p>					
17. Key Words (Suggested by Author(s)) Universal format Map Cluster file Numerical Dump Classification file Ground Truth file				18. Distribution Statement	
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 100	
22. Price*					

*For sale by the National Technical Information Service, Springfield, Virginia 22161

PRECEDING PAGE BLANK NOT FILMED

CONTENTS

Section	Page
1. SCOPE.	1-1
2. APPLICABLE DOCUMENTS	2-1
3. SYSTEM DESCRIPTION	3-1
3.1 <u>SYSTEM FLOWCHART</u>	3-1
3.2 <u>HARDWARE DESCRIPTION</u>	3-5
3.3 <u>SOFTWARE DESCRIPTION</u>	3-5
3.4 <u>FILE DESCRIPTION</u>	3-6
3.4.1 <u>INPUT DATA FILE</u>	3-6
3.4.2 USER DEFINED FILE (SYMBOL FILE NAME) (SYMBOL FILE TYPE) A	3-7
3.4.3 USER DEFINED FILE (FILE NAME) CC A	3-8
3.5 <u>SOFTWARE DESCRIPTION</u>	3-10
3.5.1 SGMAP PROGRAM	3-10
3.5.2 SOFTWARE COMPONENT NO. 1 (CRD2IN)	3-12
3.5.3 SOFTWARE COMPONENT NO. 2 (NUMB)	3-14
3.5.4 SOFTWARE COMPONENT NO. 3 (INTPAR)	3-16
3.5.5 SOFTWARE COMPONENT NO. 4 (SPMAP)	3-18
3.5.6 SOFTWARE COMPONENT NO. 5 (KUSKLA)	3-20
3.5.7 SOFTWARE COMPONENT NO. 6 (GTMAP)	3-22
3.5.8 SOFTWARE COMPONENT NO. 7 (CROPP)	3-24
4. OPERATION.	4-1
4.1 <u>OPERATING INSTRUCTION</u>	4-1
4.2 <u>COMMANDS DESCRIPTION</u>	4-1
4.2.1 START	4-2
4.2.2 SGMAP	4-2
4.2.3 END	4-2
4.3 <u>OPERATING EXAMPLE</u>	4-3

CONTENTS

Section	Page
Appendices	
A. COMMON BLOCKS.	A-1
B. PROGRAM LISTINGS	B-1
C. JOB CONTROL SOFTWARE	C-1
D. SAMPLE OUTPUT	D-1

FIGURES

Figure	Page
3.1.1 System Flow for Ground/Processed Ground Truth Input Data File. . . .	3-2
3.1.2 System Flow for Cluster/Classification Input Data File	3-3
3.1.3 SGMAP Hierarchy.	3-4

SGMAP PROGRAM

1.0 SCOPE

This document contains the description of the SGMAP Program. This program produces as output either a numeric dump or a symbolic map from an input Landsat Segment Cluster/Classification file or a digitized aircraft Ground Truth/Processed Ground Truth file.

2.0 APPLICABLE DOCUMENTS

The following documents form a part of this specification:

AD 63-2457-3308-01 Transferring Badhwar Software.

AD NAS 9-15200 Technical Memorandum Format Specifications for LACIE (Phase III)
and Accuracy Assessment Computer Data Products.

3.0 SYSTEM DESCRIPTION

3.1 SYSTEM FLOWCHART

The system flowcharts for processing ground truth and cluster/classification data files are depicted in Figures 3.1.1 and 3.1.2, respectively.

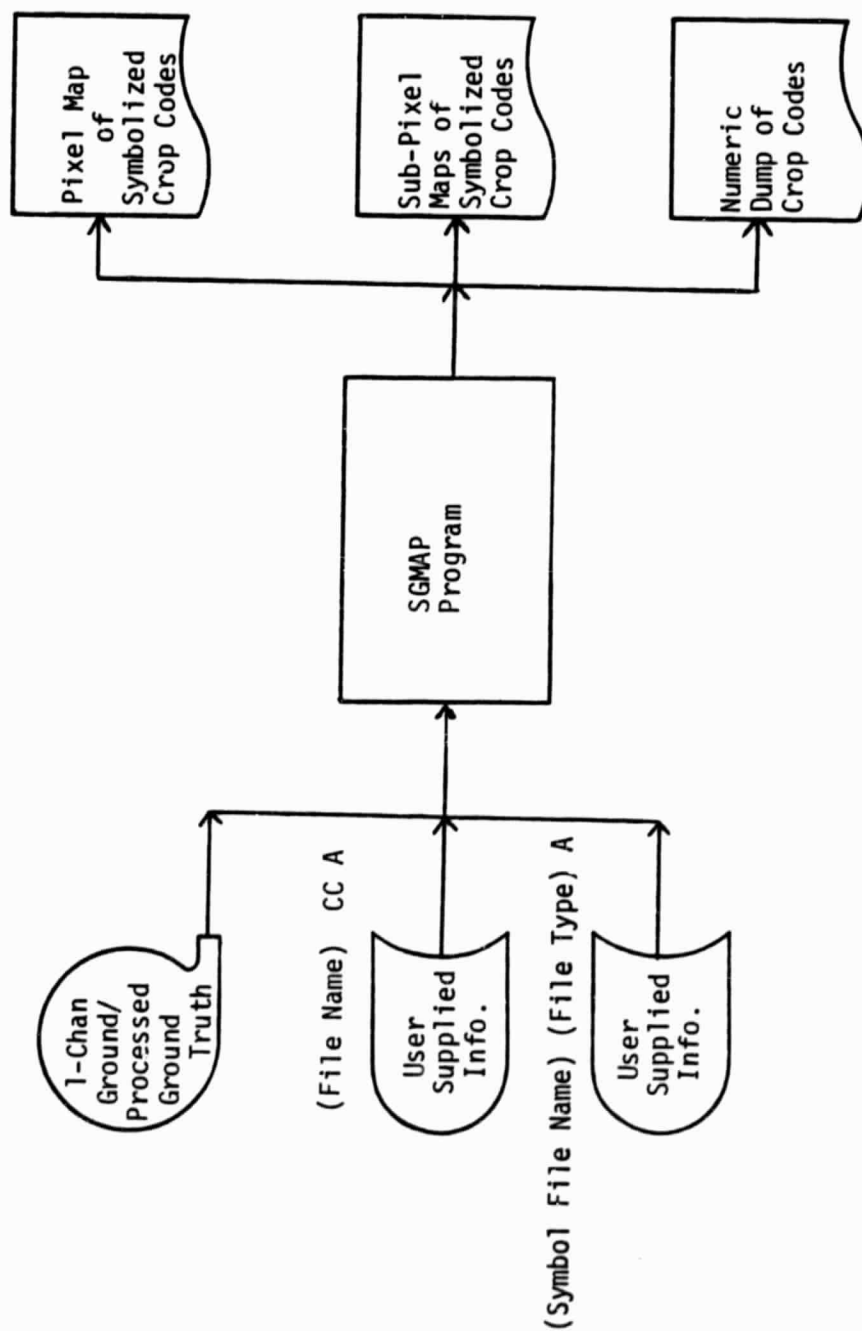


Figure 3.1.1.1 System Flow for Ground/Processed Ground Truth Input Data File

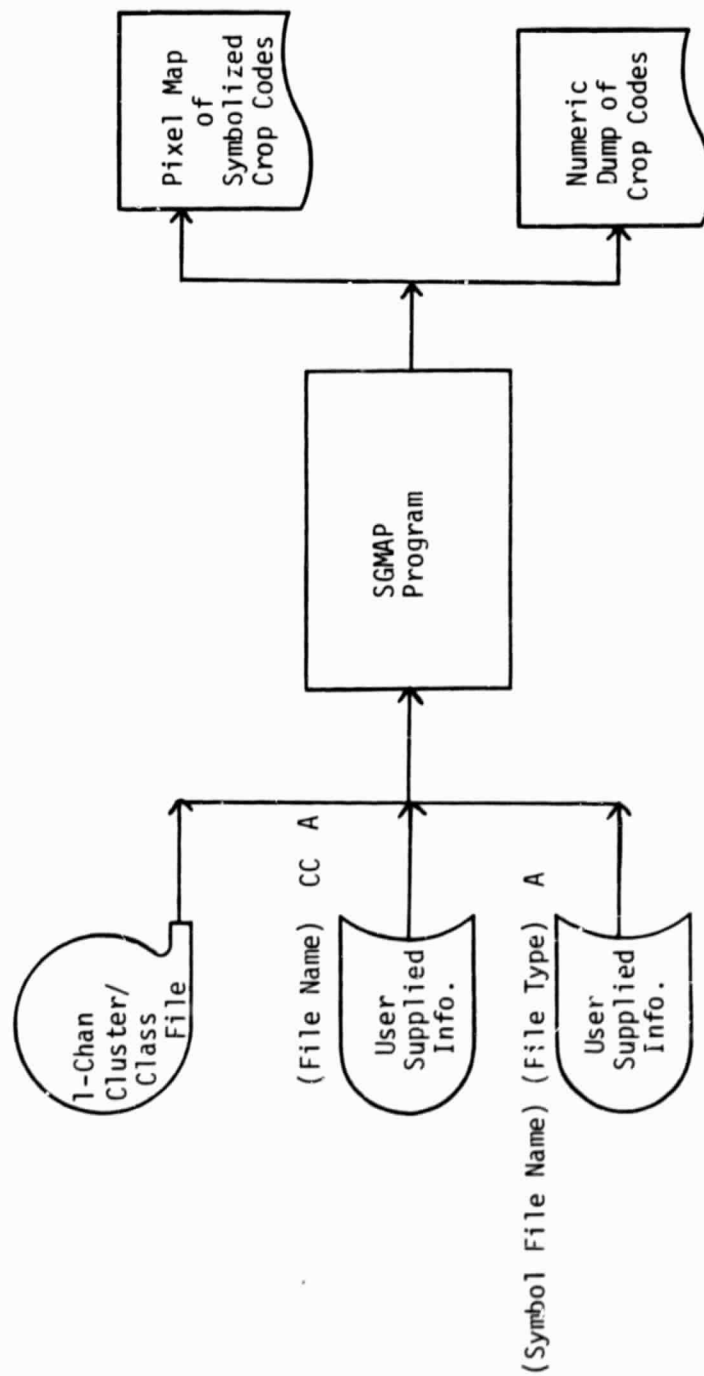


Figure 3.1.2 System Flow for Cluster/Classification Input Data File

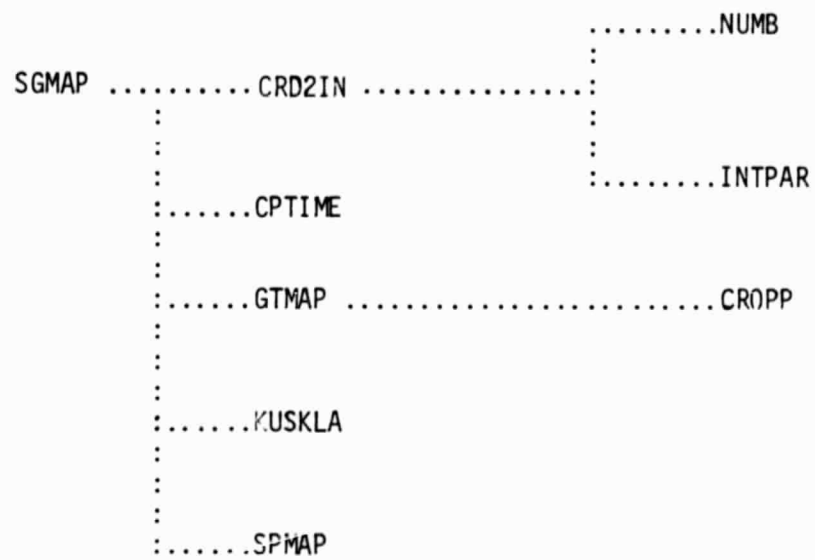


Figure 3.1.3 SGMAP Hierarchy

3.2 HARDWARE DESCRIPTION

The software for the SGMAP Program is operational on the IBM 3031 computer at Purdue.

3.3 SOFTWARE DESCRIPTION

This program produces numeric dumps or symbolic maps of two kinds of files; namely, (1) cluster or classification files and (2) ground truth files.

Cluster or classification files are single channel universal format image files produced by several different clustering and classification programs. They are "pixel level" files; i.e., they consist of 117 lines with 196 pixels per line. (See Section 3.4.1 for a complete file description).

Ground truth files are single channel universal format image files that are similar to the cluster or classification files except that they are at the sub-pixel level, i.e., they have six sub-pixels corresponding to each pixel on a cluster or classification file. They have 351 (3x117) lines with 392 (2x196) sub-pixels per line. (See Section 3.4.1 for a complete file description).

If a cluster or classification file is input, then the output numeric dump or symbolic map is necessarily at the pixel level also. However if the input is a ground truth file, then the output can be either sub-pixel level or pixel level. In the latter case the program combines the six sub-pixels that correspond to each pixel and determines the label for that pixel by majority rule. A numeric dump or symbolic map is then produced of this pixel data.

The valid range of sample values is 1-256.

3.4 FILE DESCRIPTION

3.4.1 INPUT DATA FILE

The two acceptable forms of input data are:

Cluster or Classification Files: These pixel level files are in universal format with one channel per physical record. There are 117 records, each having 360 8-bit bytes.

Ground Truth Files: These sub-pixel level files are in universal format with one channel per physical record. There are 351 records, each having 540 8-bit bytes. The contents of each byte have been biased with -128 and are stored in 8-bit two's-complement notation.

3.4.2 USER DEFINED FILE (SYMBOL FILE NAME) (SYMBOL FILE TYPE) A

For output maps, the symbol file defines a corresponding symbol for each of the 256 crop code values in the universal format input file. The user defines symbols for each crop code or crop code range; i.e, if only one crop code is mapped to a symbol, the beginning and ending of the range are the same. The usual procedure followed is that the user (1) defines all 256 codes to one symbol and (2) redefines any codes which merit unique symbols. The last entry in the file must be 0 0 0.

For output numeric dump the only entry is 0 0 N.

This file provides inputs in the given formats to the SGMAP program.

<u>Columns</u>	<u>1-5</u>	<u>6-10</u>	<u>15</u>	<u>Purpose</u>
<u>Format</u>	<u>I5</u>	<u>I5</u>	<u>A1</u>	
	start	end	symbol	Defines the start and end of a range of sample values assigned to any one symbol (MAP only).
	0	0	0	Signifies the end of SYMBOL input for a MAP.
	0	0	N	Signifies the end of SYMBOL input for a NUMERIC DUMP.

The following is an example of a symbol file for an OUTPUT Numeric Dump.

0 0 N

The following is an example of a symbol file for an OUTPUT Map.

1	256	(blank)
1	1	*
40	40	C
127	127	B
50	50	W
0	0	0

3.4.3 USER DEFINED FILE (FILE NAME) CC A

This file is used to specify a number of parameters mainly related to output description. It is composed of card records which are input to the SGMAP routine.

The first six cards have a keyword beginning in column 1 followed by parameters in columns 11 through 72. Blanks are optional. The following description lists the keywords and describes the corresponding inputs.

<u>KEYWORD</u>	<u>ACCEPTABLE INPUTS</u>	<u>DESCRIPTION</u>
TYPE	GROUND TRUTH SUB-PIXEL CLASS/CLUSTER	This card defines the type of run. In the case where the output symbolic map is a classification or cluster map, the entry on this card is "Class/Cluster". In the case where the output symbolic map is a ground truth map there are two possible entries; namely, (1) "Ground Truth" if pixel level output is desired and (2) "Sub-Pixel" if sub-pixel output is desired.
OUTPUT	MAP NUMERIC	This card specifies whether the output will be a numeric dump or a symbolic map. The corresponding entries are "Numeric" and "Map".
RECSKP ¹	Positive number	This card specifies a number of records to skip following the header record. Normally no skipping is required. However, some input files do require it. If missing, the default is 0.
DATE ¹	Free form	This card gives the date (in free-form) for report identification.
SEGMENT ¹	Segment number	This card specifies the segment number for report identification.
AI ¹	Analyst's name	This card specifies the analyst's name for report identification.
*END	Ignored	This card identifies the end of the user defined cards.

¹Optional Input

Next there are one or more cards which specify the part of the image which is to be dumped or mapped. Each card specifies the PIXEL (never sub-pixel) coordinates of the upper left corner of the area to be dumped or mapped. The output always begins at this corner and produces a display that is 117 lines by 196 samples. If pixel level output has been specified, these lines and samples are pixels; if sub-pixel output has been specified, they are sub-pixels. When pixel level output is desired, the vertex 1,1 will cause all of the image to be output. However, when sub-pixel output is desired, six vertices are required to output the entire image. Of course, fewer vertices can be used if only part of the image is desired. The vertices (line, sample) are entered one to a card and may begin in any column 1 through 72. Leading, trailing and embedded blanks on these cards are ignored. The range of values permitted is 1 through 79 for lines and 1 through 99 for samples. A maximum of six vertex cards may be used.

A \$END card is placed after the last vertex card.

All cards following the \$END card are considered comments and are printed.

The following is an example of an user's CC file where the input data is in SUB-PIXEL structure. Maps giving complete coverage of all lines and samples are desired.

```

AI          MARY ANN TOMPKINS
DATE        DECEMBER 25, 1980
SEGMENT     1981
TYPE        SUB-PIXEL
OUTPUT      MAP
*END
  1,1
 40,1
 79,1
  1,99
 40,99
 79,99
$END

```


3.5 SOFTWARE DESCRIPTION

3.5.1 SGMAP PROGRAM

Purpose

SGMAP produces either a numeric dump or a symbolic map from an input Landsat Segment Cluster/Classification file or a digitized aircraft Ground Truth/Processed Ground Truth file.

Linkages

SGMAP calls CRD2IN, CPTIME, GTMAP, KUSKLA, and SPMAP.

Interface

Calling sequence:

N/A (A description for SGMAP EXEC which can be used to load and execute SGMAP is found in Section 4.0).

Calling sequence parameters:

<u>Argument</u>	<u>Input/ Output</u>	<u>Description</u>
N/A		

Function value:

<u>Name</u>	<u>Description</u>
N/A	

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

<u>Label</u>	<u>Variable</u>	<u>Element Position</u>	<u>Input/ Output</u>	<u>Description</u>
/MAP/	IXLBUF(3060)	1	0	Pixel/subpixel input buffer.
	IXLINE(98)	2	0	Output transformed data line.
	IXLTRN(256)	3	0	Transformations for a sample value.

Blank COMMON parameters:

Full description of blank COMMON is contained in Appendix A.

<u>Variable</u>	<u>Element Position</u>	<u>Input/ Output</u>	<u>Description</u>
-----------------	-----------------------------	--------------------------	--------------------

None

Inputs

<u>Unit</u>	<u>Type</u>	<u>Description</u>
11	seq. data	Universal formatted ground truth, classification, or cluster file (See section 3.4.1).
13	seq. data	Symbolic maping information (See section 3.4.2).

Outputs

<u>Unit</u>	<u>Type</u>	<u>Description</u>
6	Report	Printout of the user defined control cards.
14	Report	Symbolic map or numeric dump of input data.

Storage requirement

Not applicable.

Description

SGMAP, according to user specification and type of input data file, produces symbolic maps or dumps in pixel or subpixel form (See Appendix C).

Flowchart

Not applicable.

Listing

See Appendix B for program listing.

3.5.2 SOFTWARE COMPONENT NO. 1 (CRD2IN)

Purpose

Subroutine CRD2IN reads, decodes, and writes to a printer file the user supplied card control file.

Linkages

CRD2IN is called by SGMAP.

CRD2IN calls INTPAR and NUMB.

Interface

Calling sequence:

CALL CRD2IN (MTYPE, KOUT, NRECSK, LFTCRS, IERR).

Calling sequence parameters:

<u>Argument</u>	<u>Input/ Output</u>	<u>Description</u>
MTYPE	0	Identifies the type of conversion of input structure to output structure.
KOUT	0	Identifies the format of output.
NRECSK	0	Number of records to skip.
LFTCRS(6)	0	Identifies the upper left corner of output.
IERR	0	Input error indicator flag.

Function value:

N/A.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.
None.

Blank COMMON parameters:

Full description of blank COMMON is contained in Appendix A.
None.

Inputs

<u>Unit</u>	<u>Type</u>	<u>Description</u>
12	seq. data	Control card file (See Section 3.4.3).

Outputs

<u>Unit</u>	<u>Type</u>	<u>Description</u>
6	Printer	File contains copy of report summary of user defined cards.
30		Used as a reread file.

Storage requirement

Not applicable.

Description

CRD2IN reads the key word (first four bytes) on the Control Card and then outputs the card. If the key word is not recognized, an error message is printed and execution continues. If the key word is recognized, the appropriate action regarding the card's parameter is taken. The routine determines if all required cards are present and if the parameters are acceptable. If either test fails, an error flag is set.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.3 SOFTWARE COMPONENT NO. 2 (NUMB)

Purpose

NUMB decodes character information into numerical information.

Linkages

NUMB is called by CRD2IN.

Interface

Calling sequence:

CALL NUMB (KHAR, NUMBER, ICOR).

Calling sequence parameters:

<u>Argument</u>	<u>Input/ Output</u>	<u>Description</u>
KHAR(80)	I	Card image.
NUMBER	O	Decoded number.
ICOL	I	Column to start decode procedure

Function value:

<u>Name</u>	<u>Description</u>
N/A	

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

<u>Label</u>	<u>Variable</u>	<u>Element Position</u>	<u>Input/ Output</u>	<u>Description</u>
None				

Blank COMMON parameters:

Full description of blank COMMON is contained in Appendix A.

<u>Variable</u>	<u>Element Position</u>	<u>Input/ Output</u>	<u>Description</u>
None			

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

Decodes a number on a user supplied control card starting with the first non-blank character in the control card parameter field (ICOL) and ending with the first non-blank, non-numeric character.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.4 SOFTWARE COMPONENT NO. 3 (INTPAR)

Purpose

Read and decode user defined control cards consisting of a pair of numbers.

Linkages

INTPAR is called by CRD2IN.

Interface

Calling sequence:

```
.. CALL INTPAR (LFTCRS, IERR).
```

Calling sequence parameters:

<u>Argument</u>	<u>Input/ Output</u>	<u>Description</u>
LFTCRS(6)	0	Identifies upper left corner of output.
IERR	0	Identifies input error IERR = 0 OK IERR = 1 Error

Function value:

N/A

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.
None.

Blank COMMON parameters:

Full description of blank COMMON is contained in Appendix A.
None.

Inputs

<u>Unit</u>	<u>Type</u>	<u>Description</u>
12	Seq. data	Control card file (See Section 3.4.3).

Outputs

<u>Unit</u>	<u>Type</u>	<u>Description</u>
6	Printer	File contains copy of report summary of user defined control cards.
30		Used as a reread file.

Storage requirement

Not applicable.

Description

INTPAR reads and decodes number pairs, input one to a card, separated by a comma.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.5 SOFTWARE COMPONENT NO. 4 (SPMAP)

Purpose

SPMAP reads input data in pixel structure and transforms the data into output form.

Linkages

SPMAP is called by SGMAP.

Interface

Calling sequence:

CALL SPMAP (PIXPRO, MOFFS, LINE, LNS).

Calling sequence parameters:

<u>Argument</u>	<u>Input/ Output</u>	<u>Description</u>
PIXPRO	O	Current number of pixels processed.
MOFFS	I	Output pixel offset.
LINE	I	Starting line of pass.
LNS	I	Current line.

Function value:

N/A.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

<u>Label</u>	<u>Variable</u>	<u>Element Position</u>	<u>Input/ Output</u>	<u>Description</u>
/MAP/	IXLBUF(3060)	1	O	Subpixel input buffer.
	IXLINE(98)	2	O	Transformed data line.
	IXLTRN(256)	3	I	Transformations for a sample value.

Blank COMMON parameters:

Full description of blank COMMON is contained in Appendix A.

None.

Inputs

<u>Unit</u>	<u>Type</u>	<u>Description</u>
11	Seq. data	Universal formatted ground truth file (See Section 3.4.1).

Outputs

None.

Storage requirement

Not applicable.

Description

SPMAP reads one line of data into an internal buffer beginning with the first requested line. Each pixel is assigned an output symbol by using the pixel value to index into the pixel transformation buffer which contains a user defined symbol. This symbol is then stored into an output buffer for eventual output.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.6 SOFTWARE COMPONENT NO. 5 (KUSKLA)

Purpose

KUSKLA reads input data in pixel structure and transforms the data into output form.

Linkages

KUSKLA is called by SGMAP.

Interface

Calling sequence:

CALL KUSKLA (MOFFS).

Calling sequence parameters:

<u>Argument</u>	<u>Input/ Output</u>	<u>Description</u>
MOFFS	I	Output pixel offset.

Function value:

N/A.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

<u>Label</u>	<u>Variable</u>	<u>Element Position</u>	<u>Input/ Output</u>	<u>Description</u>
/MAP/	IXLBUF(3060)	1	0	Pixel input buffer.
	IXLINE(98)	2	0	Transformed data line.
	IXLTRN(256)	3	I	Transformations for a sample value.

Blank COMMON parameters:

Full description of blank COMMON is contained in Appendix A.

None.

Inputs

<u>Unit</u>	<u>Type</u>	<u>Description</u>
11	Seq. data	Universal formatted classification or cluster file (See Section 3.4.1).

Outputs

None.

Storage requirement

Not applicable.

Description

KUSKLA reads one line of data into an internal buffer, transforms each pixel into an output symbol, and stores the pixel into an output buffer for eventual output.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.7 SOFTWARE COMPONENT NO. 6 (GTMAP)

Purpose

GTMAP reads input data in subpixel structure and converts the data to a pixel value.

Linkages

GTMAP is called by SGMAP.

GTMAP calls CROPP.

Interface

Calling sequence:

CALL GTMAP (PIXPRO, MOFFS).

Calling sequence parameters:

<u>Argument</u>	<u>Input/ Output</u>	<u>Description</u>
PIXPRO	0	Current number of pixels processed.
MOFFS	I	Output pixel offset.

Function value:

None.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

<u>Label</u>	<u>Variable</u>	<u>Element Position</u>	<u>Input/ Output</u>	<u>Description</u>
/MAP/	IXLBUF(3060)	1	0	Subpixel input buffer.
	IXLINE(98)	2	0	Output transformed data line.
	IXLTRN(256)	3	I	Transformations for a sample value.
/NSBIXL/	NSUBPX(6)	1		Subpixel that map to a pixel.

Blank COMMON parameters:

Full description of blank COMMON is contained in Appendix A.

N/A.

Inputs

<u>Unit</u>	<u>Type</u>	<u>Description</u>
11	Seq. data	Universal formatted ground truth file.

Outputs

None.

Storage requirement

Not applicable.

Description

GTMAP reads three lines of subpixel structure data into an internal buffer. The sub-pixels are divided into sub-pixel sets (2 sub-pixel across x 3 lines down) and CROPP is called to rule on a pixel value for the set. Each pixel is assigned an output symbol by using the pixel value to index into the pixel transformation buffer which contains a user defined symbol. This symbol is then stored into an output buffer for eventual output.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.8 SOFTWARE COMPONENT NO. 7 (CROPP)

Purpose

CROPP accepts six subpixels as input and outputs a pixel value representative of the subpixels.

Linkages

CROPP is called by GRMAP.

Interface

Calling sequence:

CALL CROPP (KROP, IXLCNT).

Calling sequence parameters:

<u>Argument</u>	<u>Input/ Output</u>	<u>Description</u>
KROP	0	Majority crop code.
IXLCNT	0	Count of number subpixels that match the majority code.

Function value:

N/A.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

<u>Label</u>	<u>Variable</u>	<u>Element Position</u>	<u>Input/ Output</u>	<u>Description</u>
/NSBIXL/	NSUBPX(6)	1	I	Subpixels that map to a pixel.

Blank COMMON parameters:

Full description of blank COMMON is contained in Appendix A.

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

The first sub-pixel value upon entry is considered to be the majority pixel (KROP) value. The NSUBPX array is traversed and the number of codes equal to KROP counted. This procedure continues until a count of each unique code is made. If at anytime a count becomes greater than the count of KROP this code becomes KROP's value.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

4.0 OPERATION

4.1 OPERATING INSTRUCTIONS

SGMAP is operational on the IBM 3031 computer at LARS, West Lafayette, Indiana.

This program requires the use of a D disk which is assigned as a temporary disk. This disk will be assigned by the SGMAP EXEC; and the user, therefore, should not assign a disk to his machine using a MODE D.

Prior to invoking the SGMAP EXEC, the user must establish on his A disk a SYMBOL file as described in section 3.4.3 and a USER CONTROL file as described in section 3.4.4.

4.2 COMMANDS DESCRIPTION

To execute SGMAP, the user enters the following series of commands which invoke the JOB CONTROL SOFTWARE.

START

SGMAP.....

END

The following sections describe each of these commands in detail. Input fields are separated by blanks. If more than one word is required to describe an input field, the description is enclosed in pointed brackets <>. If an input is optional the field is enclosed in square brackets []. Do not include these explanatory characters <> [] when actually submitting input to the computer. To enter a command, the user types one input per defined input field and separates each field with a blank.

4.2.1 START

The START command spools the user's console file. The use of this command alone with the END command will provide a listing of all information appearing on the user's console file. (If running an interactive job, this is the terminal; if running a batch job, this is a system defined device).

The START command is invoked by the user typing the following:

START

4.2.2 SGMAP

The SGMAP command executes the program. The SGMAP has the following as forms and is invoked by typing one of the following, according to the users requirement.

```
SGMAP <SYMBOL FILE> <SYMBOL FILE> <CC FILE> TAPE# FILE# [TAPE DENSITY]
      < NAME > < TYPE > < NAME >
```

If the file is on disk -

```
SGMAP <SYMBOL FILE> <SYMBOL FILE> <CC FILE> <FILENAME> <FILETYPE> <FILEMODE>
      < NAME > < TYPE > < NAME >
```

The output from the SGMAP program is spooled to the HOUSTON line printer. The output consists of a MAP and the USER IDENTIFICATION file.

4.2.3 END

This command closes the user's console file and causes a copy to be sent to the printer. This command has no effect if the START command was not previously issued. The END command is invoked by the user typing the following:

END

4.3 OPERATING EXMPLE

For our example we will assume the following:

The user's symbol file is established as required on his A disk under the file description:

880 79082 A

The user's information file is established as required on his A disk under the filename of his choice and the required filetype (CC). (For our example we will use SYM880 CC A for our complete file description.)

The user has a 1600 BPI tape# 2645 which contains the file (17) he wishes to map.

COMMAND	EXPLANATION OR ACTION TAKEN
START	Spools the console file.
SGMAP 880 79082 SYM880 2645 17	Executes the SGMAP PROGRAM The user's tape is 1600 BPI; therefore, the user inputs tape#,file#,exercising the option of using the defined default of 1600 BPI.
END	Closes the user's console file and spools the file to the HOUSTON printer.

APPENDIX A
COMMON BLOCKS

COMMON BLOCKS

COMMON/MAP/IXLBUF(3060),IXLINE(98),IXLTRN(256)
IXLBUF PIXEL/SUB-PIXEL INPUT BUFFER
IXLINE OUTPUT TRANSFORMED DATA
IXLTRN TRANSFORMATIONS FOR SAMPLE VALUE

COMMON/NSBIXL/NSUBPX(6)
NSUBPX 6 GROUND TRUTH SUB-PIXELS THAT ARE ASSOCIATED WITH
 A CORRESPONDING LACIE PIXEL

APPENDIX B
PROGRAM LISTINGS

[illegible]

SYMBOL	INTERNAL STATEMENT NUMBERS		***** OBJECTS *****		C O S S	R E F E R E N C E	L I S T I N G
	00000	00000	00000	00000			
C	0006	0006	0117	0089			
G	0006	0006	0115	0124			
T	0010	0011	0013	0013	0013		
I	0106	0106	0132	0132	0025	0047	0095 0095
J	0113	0113	0132	0132			
L	0040	0040	0053	0053	0112	0114	0114
M	0006	0006	0050	0053			
N	0006	0006	0035	0043	0112		
S	0006	0006	0021	0054	0056		
R	0029	0031	0013	0035	0037	0091	0127

NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.
C	F	I*4	0003C0	G	I*4	0003C4	0003C4	J	F	I*4	0003C4
NH	SF	I*4	0003D0	NE	SF	0003D4	0003D4	TI	SFA	I*4	0003D8
VI	SFA	I*4	0003E0	HNS	SFA	0003F4	0003F4	INSG	SFA	I*4	0003F8
DATE	SFA	I*4	000434	IEV	SFA	000438	000438	ISEG	SFA	I*4	000444
IXLS	SF	I*4	000404	KUUI	SFA	00040C	00040C	MOFE	SFA	I*4	000414
GIMAP	SF	I*4	000400	IPASS	SF	000414	000414	MTYPE	SFA	I*4	000420
SPMAP	SF	I*4	000000	CPTIME	SF	000418	000418	IMCOM#	SF	I*4	000430
IXLHIF	SF	L*1	000000	IXLNUM	SF	0004F4	0004F4	IXLTRN	SF	I*4	000440
KUSKLA	SF	R*4	000000	LFTICKS	SFA	000468	000468	PIXCNT	SF	I*4	000424
PIXPRO	SFA	R*4	00042C								

***** COMMON INFORMATION *****

NAME OF COMMON BLOCK * MAP* SIZE OF BLOCK 0017C HEXADECIMAL BYTES

VAR.	NAME	TYPE	REL.	ADDR.	VAR.	NAME	TYPE	REL.	ADDR.
IXLHIF	L*1	000000	000BF4	000000	IXLTRN	I*4	00007C	00007C	00007C

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR	ISN	ADDR	LABEL	ISN	ADDR
10	11	0004E2	29	0005C4	40	44	000750
50	46	000766	49	00077A	40	71	000A78
90	72	000A82	111	000AF0	130	122	0008D0
140	130	000C40	134	000C9H			

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	ISN	ADDR	LABEL	ISN	ADDR
100000	1	000444	12	0004FA	100007	16	00054C
100008	14	000560	21	0005H0	100011	25	000584
100012	24	00059C	40	0006F4	100015	35	000654
100016	31	00068C	49	00076A	100019	42	000738
100020	37	000740	55	0007CA	100024	50	00077E
100024	43	00077C	61	0007EE	100027	56	0007CE
100028	51	0007D6	69	00084E	100031	62	0007FC
100032	57	000800	80	0008FH	100040	75	0008A8
100036	63	000840	86	00093A	100047	81	000900
100040	67	00090C	90	0009H0	100052	87	0009A4
100044	81	000944	95	0009A6	100056	91	000980
100048	84	000934	106	0009CE	100062	100	000A26
100052	88	000998	112	000AHC	100068	107	000A9E
100056	92	000998	113	000AH2	100077	113	000B0E
100060	96	000A3C	116	000H46	100081	114	000B44
100064	102	000ACA	121	000H8C	100085	123	000B80
100068	109	000R16	127	000HC2	100089	128	000C24
100072	113	000R16					
100076	115	000R56					
100080	119	000R56					
100084	125	000RFA					

FORMAT STATEMENT LABELS

LABEL	ISN	ADDR	ISN	ADDR	LABEL	ISN	ADDR
1000	14	000028	20	000083	1006	23	000090
1010	24	0000H3	32	0000F4	1016	39	000100
1040	64	00012F	40	00013E	1052	76	000163
1070	79	000165	82	000201	1062	96	00020C
1094	94	000218	105	000224	1110	108	000232
1120	124	000234	133	000298			

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODHL(NONE)

*OPTIONS IN EFFECT*SOURCE EHCDC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(1)

STATISTICS SOURCE STATEMENTS = 134* PROGRAM SIZE = 3268* SUBPROGRAM NAME = MAIN

STATISTICS NO DIAGNOSTICS GENERATED

***** END OF COMPILATION *****

252K BYTES OF CORE NOT USED

ORIGINAL PAGE IS
OF POOR QUALITY

REQUESTED OPTIONS: NOTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(40) SIZE(MAX) AUTOHRL(MONF)

SOURCE EXECUTIVE NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NUANSF NOTERM IBM FLAG(1)

ISN 0002

SUBROUTINE CRD2INITYPE,KOUT,NRECSK,LFCRS,IERR)

INTEGER MTYPE TYPE OF OUTPUT REQUESTED
 INTEGER KOUT KIND OF OUTPUT
 INTEGER NRECSK NUMBER OF REC TO SKIP IN FILE
 INTEGER LFCRS(6,2) UPPER LEFT CORNERS(S) OF OUTPUT
 INTEGER IERR ERROR FLAG
 0 - NO ERROR(S)
 1 - ERROR(S)

HISTORY

MARY TOMPKINS LEMSCO 10/30/80 ORIGINAL CODE

PURPOSE

HEAD AND ANALYZE CARDS DESCRIBING THE FOLLOWING:
 FILE - DOCUMENTATION: LIST OF INPUT FILES
 TYPE - INPUT: FIRST NON-FLANK CHARACTER AFTER COLUMN 10

ACCEPTABLE INPUTS:
 G - GROUND TRUTH
 S - SUB PIXEL
 C - CLASS/CLUSTER

OUTPUT - INPUT: KIND OF OUTPUT
 M - MAP
 N - NUMERICAL DUMP

RECSK - INPUT: NUMBER OF RECORDS TO SKIP WITHIN FILE
 (DEFAULT 0)

DATE - DOCUMENTATION: CURRENT DATE

SEGMENT - DOCUMENTATION: SEGMENT NUMBER

AI - DOCUMENTATION: ANALYST'S NAME

*END - SPECIFY THE END OF USER DEFINED RECORDS.

EXTERNAL REFERENCES

NUMR DECODE A NUMBER
 INTPAR DECODE INTEGER PAIR(S)

EXCEPTION

1. IF ANY OF THE FOLLOWING CARDS ARE MISSING OR DATA IS UNACCEPTABLE, ISSUE A DIAGNOSTIC MESSAGE AND SET IERR = 1.
 OUTPUT TYPE
2. IF A CARD IS OF AN UNDEFINED TYPE, ISSUE A DIAGNOSTIC MESSAGE AND CONTINUE.
3. IF THERE ARE NO USER DEFINED UPPER LEFT CORNERS, ISSUE A DIAGNOSTIC MESSAGE AND SET IERR = 1.
4. IF THERE IS NO RECSK CARD THEN RECSK DEFAULTS TO 0.

LOCAL DECLARATIONS

INTEGER KARD CARD READ WORD AT A TIME
 INTEGER KOUT CARD HEAD CHARACTER AT
 INTEGER COUNT COUNT OF NUMBERS RETURNED FROM FUNCTION NUMB
 INTEGER INTRC DECODED NUMBER
 INTEGER INTRC ALLOWABLE CARD TYPES
 INTPAR KARDTY CARD TYPE

CRD000010
 CRD000020
 CRD000030
 CRD000040
 CRD000050
 CRD000060
 CRD000070
 CRD000080
 CRD000090
 CRD000100
 CRD000110
 CRD000120
 CRD000130
 CRD000140
 CRD000150
 CRD000160
 CRD000170
 CRD000180
 CRD000190
 CRD000200
 CRD000210
 CRD000220
 CRD000230
 CRD000240
 CRD000250
 CRD000260
 CRD000270
 CRD000280
 CRD000290
 CRD000300
 CRD000310
 CRD000320
 CRD000330
 CRD000340
 CRD000350
 CRD000360
 CRD000370
 CRD000380
 CRD000390
 CRD000400
 CRD000410
 CRD000420
 CRD000430
 CRD000440
 CRD000450
 CRD000460
 CRD000470
 CRD000480
 CRD000490
 CRD000500
 CRD000510
 CRD000520
 CRD000530
 CRD000540
 CRD000550
 CRD000560
 CRD000570
 CRD000580
 CRD000590
 CRD000600
 CRD000610
 CRD000620
 CRD000630
 CRD000640
 CRD000650
 CRD000660
 CRD000670
 CRD000680
 CRD000690
 CRD000700
 CRD000710
 CRD000720
 CRD000730

ORIGINAL PAGE IS
OF POOR QUALITY

ISN	CC	PROCEDURE	CRN
ISN 0003	CC	-----	CRN00740
ISN 0004	CC	INTEGER KARD(20),KHAR(80),LFLEICKS(6,2)	CRN00750
ISN 0005	CC	INTEGER INVFCT(8),TYPE*,RECS*,OUTPT*,FILE*,DATE*,SEGM*,AI*,	CRN00760
	CC	*END//	CRN00770
	CC	INTFGEH IRLANK// *	CRN00780
	CC		CRN00790
	CC		CRN00800
	CC		CRN00810
	CC		CRN00820
	CC		CRN00830
	CC		CRN00840
ISN 0006	CC	INITIALIZE INPUT VARIABLES SET IN CC FILE	CRN00850
ISN 0007	CC	MTPF = IRLANK	CRN00860
ISN 0008	CC	KOUT = IRLANK	CRN00870
ISN 0009	CC	NRECSK = 0	CRN00880
ISN 0010	CC	IfPR = 0	CRN00890
ISN 0011	CC	KOUNT = 1	CRN00900
ISN 0012	CC		CRN00910
ISN 0013	CC	*WHITE HEADING FOR INPUT SUMMARY	CRN00920
ISN 0014	CC	WHITE(6,900)	CRN00930
ISN 0015	CC	FORMAT(1,1) INPUT SUMMARY(1)	CRN00940
ISN 0016	CC		CRN00950
ISN 0017	CC		CRN00960
ISN 0018	CC		CRN00970
ISN 0019	CC		CRN00980
	CC	HEAD CARD, PRINT CARD ON SUMMARY, PLACE CARD IN REREAD BUFFER	CRN00990
	CC	HEAD(12,1000,END=100)(KARD(1),I = 1,20)	CRN01000
	CC	FORMAT(20A4)	CRN01010
	CC	REWIND 30	CRN01020
	CC	WRITE(30,1000)(KARD(1),I = 1,20)	CRN01030
	CC	REWIND 30	CRN01040
	CC		CRN01050
	CC	WHITE(6,1001)(KARD(1),I = 1,20)	CRN01060
	CC	FORMAT(1X,20A4)	CRN01070
	CC		CRN01080
	CC	ANALYZE CARD	CRN01090
	CC	-- DETERMINE CARD TYPE	CRN01100
	CC		CRN01110
	CC	DO 20 I = 1,4	CRN01120
ISN 0020	CC	KARDY = KARD(I)	CRN01130
ISN 0021	CC	IF(KARDY.EQ.INVEC(I)) GO TO(30,40,50,60,60,60,70),I	CRN01140
ISN 0022	CC		CRN01150
ISN 0024	CC	20 CONTINUE	CRN01160
	CC		CRN01170
	CC	INVALID CARD TYPE	CRN01180
	CC		CRN01190
ISN 0025	CC	WHITE(6,1002)	CRN01200
ISN 0026	CC	FORMAT(1,1) INVALID INPUT CARD -- IGNORED*	CRN01210
ISN 0027	CC	GO TO 10	CRN01220
	CC		CRN01230
	CC	TYPE CARD	CRN01240
	CC		CRN01250
ISN 0028	CC	30 READ(30,1003)(KHAR(1),I = 1,80)	CRN01260
ISN 0029	CC	FORMAT(8A1)	CRN01270
	CC		CRN01280
	CC	DO 35 I = 1,72	CRN01290
ISN 0030	CC	IF(KHAR(1),NE.IRLANK) MTYPE = KHAR(I)	CRN01300
ISN 0031	CC	IF(KHAR(1),NE.IRLANK) GO TO 10	CRN01310
ISN 0032	CC		CRN01320
ISN 0033	CC	CONTINUE	CRN01330
ISN 0034	CC	GO TO 10	CRN01340
ISN 0035	CC		CRN01350
ISN 0036	CC		CRN01360
	CC		CRN01370
	CC	RECSKP CARD	CRN01380
	CC		CRN01390
ISN 0037	CC	40 READ(30,1003)(KHAR(1),I = 1,80)	CRN01400
ISN 0038	CC		CRN01410
ISN 0039	CC	ICOL = 11	CRN01420
ISN 0040	CC	KOUNT = NUMR(KHAR,NUMR,ICOL)	CRN01430
ISN 0041	CC	NRECSK = NUMBER	CRN01440
	CC	GO TO 10	CRN01450
	CC		CRN01460
	CC	OUTPUT CARD	CRN01470
	CC		CRN01480
	CC		CRN01490
	CC		CRN01500
	CC		CRN01510

ORIGINAL PAGE IS
OF POOR QUALITY

PAGE 3

DATE 81.139/13.19.55

FORTHAN H EXTENDED

05/360

CRO2IN

(JUNE 78)

*LEVEL 2.3.0

```

C 50 READ(30,1003) (KHAR(I),I = 1,80)
C 100 55 I = 11,72
C IF (KHAR(I).NE.IHLANK) KOUT = KHAR(I)
C IF (KHAR(I).NE.IHLANK) GO TO 10
C 55 CONTINUE
C GO TO 10
C
C DOCUMENTARY CARDS: FILE,DATE,SEGMENT,AI
C 60 GO TO 10
C
C *END CARD = CHECK USER DEFINED UPPER LEFT CORNER OF OUTPUT
C 70 CALL INTPAR(LFICRS,IERR)
C
C CHECK FOR COMMENTS
C 80 READ(12,1000,FNO=100) (KARD(I),I = 1,20)
C WRITE(6,1001) (KARD(I),I = 1,20)
C GO TO 80
C
C CHECK FOR REQUIRED INPUTS
C 100 IF (KOUNT.LT.0) WRITE(6,1004) NRECSK
C 1004 FORMAT(' INVALID RECSKP CARD ',I4,' NOT ACCEPTABLE.')
C IF (KOUNT.LE.0) IERR = 1
C
C 1005 IF (MYPE.EQ.IHLANK) WRITE(6,1005)
C 1005 FORMAT(' TYPE CARD MISSING.')
C IF (MYPE.EQ.IHLANK) IERR = 1
C
C 1006 IF (KOUT.EQ.IHLANK) WRITE(6,1006)
C 1006 FORMAT(' OUTPUT CARD MISSING.')
C IF (KOUT.EQ.IHLANK) IERR = 1
C
C 1007 IF (LFICRS(I).EQ.0) WRITE(6,1007)
C 1007 FORMAT(' DEFINED AT LEAST ONE UPPER LEFT CORNER OF OUTPUT.')
C IF (LFICRS(I).EQ.0) IERR = 1
C RETURN
C END

```

***** LIST IN G*****

SYMBOL	INTERNAL STATEMENT NUMBERS	CROSS REFERENCE	LIST IN G
I	0013	0018	0031 0033
ICOL	0037	0016 0018 0043	0030 0053
IERR	0038	0042 0044 0046 0052 0053	0028 0052
KARD	0002	0051 0058 0063 0068 0073	
KHAR	0003	0016 0018 0021 0052 0053	
KOUT	0004	0007 0044 0065	
NUMH	0039	0004 0044 0060 0063 0065 0068	
INVEC	0004	0022 0058	
KOUNT	0010	0039 0055	
MYPE	0006	0031 0060 0063	
CRO2IN	0002		
IHLANK	0005	0006 0007 0031 0033 0044 0046 0060 0063 0065 0068	
INTPAR	0051		
KARDTY	0021		
LFICRS	0002	0051 0070 0073	
NRECSK	0002	0040 0055	
NUMBER	0039		

***** LIST IN G*****

LABEL	DEFINITION	CROSS REFERENCE	LIST IN G
10	0013		
20	0027		
30	0028		
35	0029		
40	0030		
50	0031		
55	0032		

LABEL	DEFINED	***** REFERENCES	CROSS	REFERENCE	***** LISTING
10	0010	0007			
20	0017	0005 0011			

REQUESTED OPTIONS: NONE

OPTIONS IN EFFECT: NAME(MAIN), OPTIMIZE(1), LINECOUNT(80), SIZE(MAX), AUTOJUL(NONE),
SOURCE FRODIC, HOLIST, NODECK, OBJECT MAP, NOFORMAT, NOGOSTMT, XHEF, ALC, NOANSF, NOTERM, IRM, FLAG(1)

ISN 0002

SUBROUTINE GTMAP(PIXPRO,MOFFS)

REAL PIXPRO PIXEL PROCESSED
INTEGER MOFFS OUTPUT OFFSET

HISTORY

CARL AKLFPS LEC ORIGINAL CODE (PDP)
MARY TOMPKINS LFMSCO 11/21/80 REDD (IBM)

PURPOSE

THIS ROUTINE PROCESSES PIXEL MAPS FROM UNIVERSAL TAPES IN
SUB-PIXEL FORM (2X196 SUB-PIXELS X 3X117 LINES). THE CROP
CODES HAVE BEEN HIAS -12H AND ARE EITHER NUMERIC IN VALUE
OR SYMBOLIC ACCORDING TO USER SPECIFIED OPTIONS.

EXTERNAL REFERENCES

CROPP EVALUATES SUB-PIXEL CROP CODES FOR PIXEL CODE

EXCEPTIONS

NONE.

LOCAL DECLARATIONS

INTEGER LINFND POINTER TO LINE END IN BUFFER
INTEGER IXLADJ PIXEL # ADJUSTMENT FACTOR (ADVANCE FOR
 EACH PAIR OF PIXELS)
INTEGER IXLCAN SUB-PIXEL CANDIDATE FOR PIXEL LINE
INTEGER IXLCT2 PIXEL COUNT 2 SUB-PIXEL PER LINE
INTEGER IXLCT3 PIXEL COUNT 3 LINES OF 2 SUBS PER PIXEL
INTEGER KROPVL CROP CODE
INTEGER KROPSL CURRENT OR LAST VALID CROP CODE
INTEGER NSUBPX(6) SUB-PIXELS TO DECIDE PIXEL
 CROP CODE SELECTED

PROCEDURE

ISN 0003
ISN 0004
ISN 0005

LOGICAL*1 IXLHUF
COMMON/NSBIXL/NSUBPX(6)
 IXLHUF(3060), IXLIN(98), IXLTRN(256)

ISN 0006
ISN 0007
ISN 0008
ISN 0009
ISN 0010

HEAD 3 LINES INTO BUFFER
DO 20 L = 1, 1620, 540
 LINFND = L - 539
 BEAD(1), 1000, END=900) (IXLHUF(K), K = L, LINFND)
1000 FORMAT(13(1RM1))
20 CONTINUE

ISN 0011

IXLADJ = -470 + MOFFS

ISN 0012

LOOP FOR EACH PIXEL IN LINE
KROPVL = 1

GT-00010
GT-00020
GT-00030
GT-00040
GT-00050
GT-00060
GT-00070
GT-00080
GT-00090
GT-00100
GT-00110
GT-00120
GT-00130
GT-00140
GT-00150
GT-00160
GT-00170
GT-00180
GT-00190
GT-00200
GT-00210
GT-00220
GT-00230
GT-00240
GT-00250
GT-00260
GT-00270
GT-00280
GT-00290
GT-00300
GT-00310
GT-00320
GT-00330
GT-00340
GT-00350
GT-00360
GT-00370
GT-00380
GT-00390
GT-00400
GT-00410
GT-00420
GT-00430
GT-00440
GT-00450
GT-00460
GT-00470
GT-00480
GT-00490
GT-00500
GT-00510
GT-00520
GT-00530
GT-00540
GT-00550
GT-00560
GT-00570
GT-00580
GT-00590
GT-00600
GT-00610
GT-00620
GT-00630
GT-00640
GT-00650
GT-00660
GT-00670
GT-00680
GT-00690
GT-00700
GT-00710
GT-00720
GT-00730

ORIGINAL PAGE IS
OF POOR QUALITY

NAME		TAG		TYPE		ADDR		NAME		TAG		TYPE		ADDR		NAME		TAG		TYPE		ADDR	
KROP	CF	1	CF	1	CF	0000CC	000000	ADJ	000000	1	CF	1	CF	000000	000000	ADJ	000000	1	CF	1	CF	000000	000000
IRCOM	CF	1	CF	1	CF	000000	000000	CRPP	000000	1	CF	1	CF	000000	000000	GTMAP	000000	1	CF	1	CF	000000	000000
IXLCNT	CF	1	CF	1	CF	000000	000000	IXLADJ	000000	1	CF	1	CF	000000	000000	IXLCNT	000000	1	CF	1	CF	000000	000000
IXLTRN	CF	1	CF	1	CF	000000	000000	IXLCI2	000000	1	CF	1	CF	000000	000000	IXLCI3	000000	1	CF	1	CF	000000	000000
NSURPX	S	1	S	1	S	000000	000000	KRPPSL	000000	1	CF	1	CF	000000	000000	KRPPVL	000000	1	CF	1	CF	000000	000000
								PIXPPO	000000	1	CF	1	CF	000000	000000	LINE	000000	1	CF	1	CF	000000	000000

***** COMMON INFORMATION *****

NAME OF COMMON BLOCK * MAP* SIZE OF BLOCK 00117C HEXADECIMAL BYTES
 VAR. NAME TYPE REL. ADDR. VAR. NAME TYPE REL. ADDR. VAR. NAME TYPE REL. ADDR.
 IXLRIE L01 000000 IXLRIE 000000 IXLRIE 000000

NAME OF COMMON BLOCK * NSPIXL* SIZE OF BLOCK 00001B HEXADECIMAL BYTES
 VAR. NAME TYPE REL. ADDR. VAR. NAME TYPE REL. ADDR. VAR. NAME TYPE REL. ADDR.
 NSURPX L01 000000 NSURPX 000000 NSURPX 000000

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
50	10	000190	90	32	00023E	100	35	000260
900	36	00026A						

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
100001	1	00012C	100003	17	00014C	100014	8	00016C
100002	2	00013C	100004	18	00015C	100015	9	00017C
100003	3	00014C	100005	19	00016C	100016	10	00018C
100004	4	00015C	100006	20	00017C	100017	11	00019C
100005	5	00016C	100007	21	00018C	100018	12	00020C
100006	6	00017C	100008	22	00019C	100019	13	00021C
100007	7	00018C	100009	23	00020C	100020	14	00022C
100008	8	00019C	100010	24	00021C	100021	15	00023C
100009	9	00020C	100011	25	00022C	100022	16	00024C
100010	10	00021C	100012	27	00024C			

FORMAT STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR
1000	9	00002A			

- *OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODHL(NONE)
- *OPTIONS IN EFFECT*SOURCE ERCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(1)
- *STATISTICS* SOURCE STATEMENTS = 36, PROGRAM SIZE = 706, SUBPROGRAM NAME = GTMAP
- *STATISTICS* NO DIAGNOSTICS GENERATED

***** END OF COMPILATION *****

292K BYTES OF CORE NOT USED

ORIGINAL PAGE IS
OF POOR QUALITY

REQUESTED OPTIONS: NOTRIM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(R0) SIZE(MAX) AUTODRL(NONE)
SOURCE EHCDC NOLIST MODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(1)

TSN 0002

```
-----
SUBROUTINE INTPAR(LFICRS,IFERR)
-----
      INTEGER LFICRS(LINROW,ICOL)
      INTEGER IFERR
      LEFT CORNER(S) OF OUTPUT
      ERROR FLAG
      0 - NO ERROR(S)
      1 - ERROR(S)

      HISTORY
      -----
      MARY TOMPKINS      LEMSCU      10/27/80      ORIGINAL CODE

      PURPOSE
      -----

      READ AND INTERPHET CARDS CONSISTING OF A PAIR OF NUMBERS SEPARATED
      BY COMMAS.

      EXTERNAL REFERENCES
      -----

      NONE.

      EXCEPTIONS
      -----
      IF A NON-NUMERIC CHARACTER OTHER THAN A BLANK WHICH IS
      IGNORED OR A COMMA WHICH IS USED AS A SEPARATOR IS
      FOUND A MESSAGE IS ISSUED AND IFERR = 1.

      LOCAL DECLARATIONS
      -----
      INTEGER KARD
      INTEGER KMAP
      INTEGER TYPE
      INTEGER KOUT
      INTEGER KUNIT
      INTEGER KNUMBER
      INTEGER INT(J)

      CARD READ WORD AT A TIME
      CARD READ CHARACTER AT A TIME
      TYPE OF OUTPUT REQUESTED
      KIND OF OUTPUT
      POWER OF 10 UNIT
      DIGIT VALUE OF CHARACTER
      CHARACTER DIGITS

      INTEGER KARD(20), ICHAR(R0), LFICRS(6,2)
      INTEGER INT(10), J(0,1,2,3,4,5,6,7,8,9,99)
      DATA IBLANK/' ',KOMMA/,',',INDO/SEND/,

      ZERO FILL LEFT CORNER ARRAY
      DO 15 I = 1,6
      DO 10 J = 1,2
      LFICRS(I,J) = 0
      10 CONTINUE
      15 CONTINUE

      READ CARD AND PLACE CARD IN REREAD HUFFER
      DO 200 NROW = 1,6
      READ(12,1000,FEND=900) (KARD(I),I = 1,20)
      FORMAT(20A4)
      PERIOD 30
      MULTIPLY(30,1000) (KARD(I),I = 1,20)
      REMIND 30
      200 CONTINUE

      IF CARD IS NOT SEND ASSUME IT CONTAINS A PAIR OF NUMBERS
```

INT00010
INT00020
INT00030
INT00040
INT00050
INT00060
INT00070
INT00080
INT00090
INT00100
INT00110
INT00120
INT00130
INT00140
INT00150
INT00160
INT00170
INT00180
INT00190
INT00200
INT00210
INT00220
INT00230
INT00240
INT00250
INT00260
INT00270
INT00280
INT00290
INT00300
INT00310
INT00320
INT00330
INT00340
INT00350
INT00360
INT00370
INT00380
INT00390
INT00400
INT00410
INT00420
INT00430
INT00440
INT00450
INT00460
INT00470
INT00480
INT00490
INT00500
INT00510
INT00520
INT00530
INT00540
INT00550
INT00560
INT00570
INT00580
INT00590
INT00600
INT00610
INT00620
INT00630
INT00640
INT00650
INT00660
INT00670
INT00680
INT00690
INT00700
INT00710
INT00720
INT00730

ORIGINAL PAGE IS
OF POOR QUALITY

PAGE 2

DATE 81.139/13.25.09

05/360 FORTTRAN EXTENDED

*LEVEL 2.3.0 (JUNE 78)

INTPAR

```
ISN 0017 C
ISN 0019
ISN 0020 1001
ISN 0021
ISN 0022
C
C
C
C
C
C
ISN 0023
ISN 0024
ISN 0026
C
C
ISN 0028
ISN 0029
ISN 0030
C
C
C
C
ISN 0031 20
ISN 0033
ISN 0034 1002
ISN 0035
ISN 0036
C
C
C
C
ISN 0037 40
ISN 0038
ISN 0039
ISN 0041
ISN 0042
ISN 0043
ISN 0044
ISN 0045
ISN 0046
ISN 0047

IF (KARD(I).EQ.IND) GO TO 900
READ(36,1001) (ICOL(I),I = 1,80)
FORMAT(80A1)
ICOL = 1
IUNIT = 1

EXAMINE EACH CHARACTER. THE FIRST NUMBER STARTS WITH THE FIRST
NON-BLANK NUMERIC CHARACTER AND ENDS WHEN A COMMA IS FOUND. THE
SECOND NUMBER STARTS WITH THE FIRST NON-BLANK NUMERIC CHARACTER
ENDS WHEN THE LAST NUMERIC CHARACTER IS FOUND

DO 100 I = 1,72
  IF (ICOL(I).EQ.IND) GO TO 100
  IF (ICOL(I).NE.KOMMA) GO TO 20
  CHAR IS A COMMA SECOND NUMBER FOLLOWS.
  IUNIT = 1
  ICOL = 2
  GO TO 100

CHECK FOR NON-NUMERIC CHARACTERS
20 IF (ICOL(I).GE.INT(1).AND.ICOL(I).LE.INT(10)) GO TO 40
WRITE(6,1002)
FORMAT(1X,' ERROR ON CARD DEFINING UPPER LEFT CONNER OF ',
'OUTPUT')
IFERR = 1
GO TO 900

CHARACTER IS NUMERIC
DO 50 J = 1,10
  NUMBER = J - 1
  IF (ICOL(I).EQ.INT(J)) GO TO 60
  CONTINUE
  LFCRS(NROW,ICOL) = LFCRS(NROW,ICOL)*IUNIT + NUMBER
  IUNIT = 10
  CONTINUE
200 CONTINUE
900 RETURN
END
```

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

SYMBOL	INTERNAL STATEMENT NUMBERS	CROSS REFERENCE	LISTING
I	0007	0015	0031
IND	0008	0012	0026
INT	0009	0015	0031
ICOL	0017	0031	0039
ICOL	0021	0024	0042
KARD	0003	0015	0017
NROW	0003	0012	0015
ICOL	0011	0042	0026
IUNIT	0003	0024	0042
KOMMA	0005	0026	0043
IND	0005	0026	0043
INTPAR	0002	0008	0042
LFCRS	0002	0008	0042
NUMBER	0038	0003	0042

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

LABEL	DEFINED	REFERENCES
10	0009	0007
15	0010	0006
20	0031	0026
30	0037	0031
50	0041	0037
60	0042	0039
100	0045	0024
200	0046	0030
900		0017

ORIGINAL PAGE IS
OF POOR QUALITY

REQUESTED OPTIONS: NONE**

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODIAL(NONE)
SOURCE ERDIOC NOIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(1)

ISN 0002

SUBROUTINE KUSKLA(MOFFS)

INTEGER MOFFS OUTPUT OFFSET

HISTORY

CARL AKLERS LEC 11/21/80 ORIGINAL CODE (PDP)
MARY TOMPKINS LEMSCO REDO (IBM)

PURPOSE

REF AND CODE CROP TRANSFORMATION SYMBOLS OR NUMERIC
VALUES OF PIXEL MAPS FROM UNIVERSAL TAPES IN PIXEL FORM
(196 X 117).

EXTERNAL REFERENCES

NONE.

EXCEPTIONS

NONE.

LOCAL DECLARATIONS

INTEGER KALRYT INTERNAL TAPE CALBRATION BYTE + OFFSETS
INTEGER IXLNUM PIXEL NUMBER

PROCEDURE

LOGICAL I1 IXLBUF
COMMON/MAP/TALBUF (3060), IXLIN (98), IXLTRN (256)

HEAD ONE LINE OF SEGMENT

READ(11,1000)(IXLBUF(I),I = 1,360)
1000 FORMAT(2(190A1))

GET PIXEL TRANSFORMATION FOR OUTPUT

KALRYT = 72 + MOFFS
DO 100 I = 1,98
IXLNUM = KALRYT + I
IXLINE(I) = IXLTRN(IXLBUF(IXLNUM))
IF (IXLBUF(IXLNUM).EQ.0) IXLIN(I) = 0
100 CONTINUE
RETURN
END

ISN 0003
ISN 0004

ISN 0005
ISN 0006

ISN 0007
ISN 0008
ISN 0009
ISN 0010
ISN 0011
ISN 0012
ISN 0013
ISN 0014
ISN 0015

*****FORTRAN CROS5 REFERENCE LISTING*****

SYMBOL INTERNAL STATEMENT NUMBERS 0009 0010 0011
I MOFFS 0005 0007
IXLBUF 0002 0004 0005 0010 0011
IXLINE 0003 0004 0005 0010 0011
IXLNUM 0009 0010 0011
IXLTRN 0004 0010

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N *****

SYMBOL INTERNAL STATEMENT NUMBERS

KALRYT 0007 0009
KUSKLA 0002

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N *****

LAREL DEFINED REFERENCES
1000 0006
1000 0005

/ KUSKLA / SIZE OF PROGRAM 000186 HEXADECIMAL BYTES

NAME	TYPE	ADD.	NAME	TYPE	ADD.	NAME	TYPE	ADD.	NAME	TYPE	ADD.
IXLINE S	C	0000A4	MOFFS F	I*4	0000A4	IXLNUM SF	I*4	0000A4	IXLNUM SF	I*4	0000A4
KUSKLA	C	0000F4	IXLNUM SF	I*4	0000F4	IXLNUM SF	I*4	0000F4	IXLNUM SF	I*4	0000F4

***** COMMON INFORMATION *****

NAME OF COMMON BLOCK * MAP* SIZE OF BLOCK 00117C HEXADECIMAL BYTES

VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.
IXLNUM	L*1	000000	IXLNUM	I*4	0000F4	IXLNUM	I*4	0000F4

SOURCE STATEMENT LABELS

LAREL ISN ADDR
100 13 000146

COMPILER GENERATED LABELS

LAREL ISN ADDR
100001 2 0000CA

FORMAT STATEMENT LABELS

LAREL ISN ADDR
1000 6 000024

NUMBER LEVEL

IFF5261 4(W) ISN 0011

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODRL(NONE)

*OPTIONS IN EFFECT*SOURCE ERCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(1)

STATISTICS SOURCE STATEMENTS = 14, PROGRAM SIZE = 390, SUBPROGRAM NAME =KUSKLA

STATISTICS 1 DIAGNOSTICS GENERATED, HIGHEST SEVERITY CODE IS 4

***** END OF COMPIATION *****

292K BYTES OF CORE NOT USED

ORIGINAL PAGE IS
OF POOR QUALITY

REQUESTED OPTIONS: NOTERM

```

OPTIONS IN EFFECT: NAME(MATR) OPTIMIZE(1) LTRFCOUNT(R0) SIZE(MAX) AUTOHDL(NODE)
SOURCE FHCFCIC NOLIST NODECK OBJECT MAP NOFORMAT 'NOGOSTMT XREF ALC NOANSF NOTERM ILM FLAG(I)

```

ISSN 0002

FUNCTION NUMBER (KHAP, NUMBER, ICOL)

INTEGER KHAR INTEGER NUMBER	CARD READ CHARACTER AT A TIME DECODED NINTEH COLUMN IN CARD TO START DECODE
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	20
21	21
22	22
23	23
24	24
25	25
26	26
27	27
28	28
29	29
30	30
31	31
32	32
33	33
34	34
35	35
36	36
37	37
38	38
39	39
40	40
41	41
42	42
43	43
44	44
45	45
46	46
47	47
48	48
49	49
50	50
51	51
52	52
53	53
54	54
55	55
56	56
57	57
58	58
59	59
60	60
61	61
62	62
63	63
64	64
65	65
66	66
67	67
68	68
69	69
70	70
71	71
72	72
73	73
74	74
75	75
76	76
77	77
78	78
79	79
80	80
81	81
82	82
83	83
84	84
85	85
86	86
87	87
88	88
89	89
90	90
91	91
92	92
93	93
94	94
95	95
96	96
97	97
98	98
99	99
100	100

HISTORY

LIBRARY	TOMPKINS	L.F.M.S.C.	10/31/80	ORIGINAL CODEF

PURPOSE

DECODES A NUMBER STARTING WITH ICOL AND TERMINATING WITH THE FIRST NON-BLANK NON-NUMERIC CHARACTER.

EXTERNAL REFERENCES

ONE.

EXCEPTIONS

1. IF NO VALID NUMBER IS FOUND NUMH = 0.

LOCAL DECLARATIONS

INTEGER INT(10)	NUMERIC CHARACTERS
INTEGER IUNIT	POWER OF 10 UNIT
INTEGER NDIGIT	NEW DIGIT

PROCEDURE

```
INTEGER KHAR(H0),IBLANK//
INTEGER INT(10)//0,1,2,3,4,5,6,7,8,9,99/
```

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n \mathbb{E}[\log \frac{1}{p_i}] = 0$$

oop until character is non-numeric ignoring blanks.

```

000 200 I = ICOL,72
      IF (KCHAR(I).EQ.19LANK) GO TO 200
      IF (KCHAR(I).LT.INT(1).OR.KCHAR(I).GT.INT(10)) GO TO 900

```

```

DO 100 J = 1, 10
  N0IGIT = J-1
  IF (KCHAR(I).EQ.INT(J)) GO TO 60

```

ISN 0003
ISN 0004

NSI	9000
NSI	5000

7060102345789012
000101010010101010
000000000000000000
ZZZZZZZZZZZZZZZZZZ

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

SYMBOL	INTERNAL STATEMENT	NUMBERS
I	0007 0008 0010	0010 0014
J	0012 0013 0014	

L I S T I N : 000000

SYMBOL	INTERNAL	STATEMENT	NUMBERS	OF	CHARACTERS
IN	0004	0007	0010	0010	0014
ICOL	0003	0007	0008	0010	0014
KHAR	0002	0004	0008	0010	0014
NUMH	0002	0005	0017	0019	0014
TUNIT	0005	0007	0014	0014	0014
IBLANK	0003	0003	0004	0004	0014
NOTGAT	0013	0017	0017	0017	0014
NUMBER	0002	0017	0017	0017	0014

LIST IN (000000)

LABEL	DEFINED	REFERENCES	*****	C O R T R A N
60	0017	0014		
100	0016	0012		
200	0020	0007	000H	
900	0021	0010		

/	NUMH /	SIZE OF PROGRAM	0001F ₄	HEXADECIMAL BYTES		
TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME
I ₈₄	000094	INI		I ₈₄	0000BU	ICOL F
I ₈₄	00009C	TUNIT	SF	I ₈₄	0000d0	IHLANK
I ₈₄	0000AC					

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR
100	16	000162	200	20	000178
			60	17	000164
			900	21	000182

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
000001	2	0000F0	100002	8	00010F	100003	10	000120
000002	3	0000F0	100004	9	00010F	100005	11	000132

*OPTIONS IN FFFCT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(B0) SIZE(MAX) AUTODRL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSMT XREF ALC NOANSE NOTERM IBM FLAG(I)

STATISTICS SOURCE STATEMENTS =

•STATISTICS• NO DIAGNOSTICS GENERATED

***** FND OF COMPIATION *****

284K BYTES OF CORE NOT USED

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

PAGE 1

DATE 81.139/13.30.12

05/360 FORTRAN H EXTENDED

LEVEL 2.3.0 (JUNE 75)

REQUESTED OPTIONS: NONE

OPTIONS IN EFFECT: NAME (MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE (MAX) AUTOHPL (NONE)
SOURCE FRCDIC MOLLIST MODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(1)

ISN 0002 SUBROUTINE SPMAP(PIXPRO,MOFFS,LINE,LNS)-----

REAL PIXPRO
INTEGER MOFFS
INTEGER LINE
INTEGER LNS
PIXELS PROCESSED
OUTPUT OFFSET
STARTING LINE
CURRENT LINE

HISTORY

CARL AKERS LFC ORIGINAL CODE (PDP)
MARY TOMPKINS LMSCO 11/21/80 REDO (IBM)

PURPOSE

THIS ROUTINE PROCESSES SUB-PIXEL MAPS FROM UNIVERSAL TAPES
IN SH-PLOT FORM (2X196 SUB-PIXELS X 3X117 LINES). THE CROP
COMES HAVE BEEN BIASED -128 AND ARE EITHER NUMERIC IN VALUE
OR SYMBOLIC ACCORDING TO USER SPECIFIED OPTIONS.

EXTERNAL REFERENCES

NONE

EXCEPTIONS

NONE

LOCAL DECLARATIONS

INTEGER IXLN
INTEGER KALRYT
INTEGER IXLNUM
INTEGER KROPL
INTEGER KROPLV
LINE REFERS TO (196X117) SEGMENT
FOR (2X196 X 3X117) SEGMENT
CALCULATION INFORMATION + OFFSET
CURRENT PIXEL NUMBER
CROP CODE
CURRENT OR LAST VALID CROP CODE

PROCEDURE

LOGICAL *1 IXLHUF
COMMON/MAP/IXLHUF(3060), IXLN(98), IXLTRN(256)

STARTING LINE IS NOT 1 SO FIND IT

IF (LNS.NF.1.OR.LINE.EQ.1) GO TO 100

IXLN = (LINE - 1)*3

DO 10 I = 1, IXLN
HEAD(11,1000)(IXLHUF(J),J = 1,540)
FORMAT(3(180A1))

1000

10 CONTINUE

POSITIONED AT REQUESTED LINE

100 READ(11,1000)(IXLHUF(I),I = 1,540)

SET PIXEL TRANSFORMATION FOR OUTPUT (TAPE WAS WRITTEN FOR PDP
WHICH USES THE EIGHTH BIT AS A SIGN BIT REQUIRING AN ADJUSTMENT

ISN 0012

ISN 0003
ISN 0004

ISN 0005

ISN 0007
ISN 0008
ISN 0009
ISN 0010
ISN 0011

ORIGINAL PAGE IS
OF POOR QUALITY

SPM00740
SPM00750
SPM00760
SPM00770
SPM00780
SPM00790
SPM00800
SPM00810
SPM00820
SPM00830
SPM00840
SPM00850
SPM00860
SPM00870
SPM00880
SPM00890
SPM00900
SPM00910
SPM00920
SPM00930
SPM00940
SPM00950

L I S T I N G

REFERENCES

CRUSS

UNITED STATES

ALN	0007	000A	0018	0019	0019	0021	0023
ALP	0007	000A	0007				
INF	0002	0005					
OFFS	0002	0013					
CPMAP	0002			0012	0017		
XLHRUF	0003	0004	0009				
XLINF	0004	0025					
XLNUM	0004	0017					
XLTRN	0004	0025					
ALALRYT	0013	0016					
ROPVL	0014	0023	0025				
XLPRO	0002	0021	0021				

REFERENCE

CROSS

***** U R T

SIZE OF PROGRAM 000252 HEXADECIMAL BYTES

SPMAP /

TYPE	ADD.
I#4	0000C4
R#4	0000D4
I#4	0000D8
R#4	0000E4

NAME	SF	TAG
IXLN		
SPMAP		
IXLN		
PIXP		

TYPE	ADD.
II#4	0000C0
II#4	000000
II#4	000AF4
II#4	0000E0

NAME		TAG
LNS		
MOFFS	F	C
IXLINE	S	
KRÖPVL	SF	

TYPE	ADD.
I*4	0000HC
I*4	0000CC
I*1	000000
I*4	0000DC

NAME	LINE	TAG
IXL	F	
LAUF	F	
KALHYT	SF	C
	SF	

REF
AND •
0000HH
0000CB
000000
00007C

NAME	I	SF	X
KROP	SF		
IBCOM#	F		
IXLTRN	F		

***** COMMON INFORMATION *****

NAME OF COMMON BLOCK	MAP#	SIZE OF BLOCK	00117C HEXADECIMAL BYTES
COMMON BLOCK 1	1	100	00000000
COMMON BLOCK 2	2	200	00000000
COMMON BLOCK 3	3	300	00000000
COMMON BLOCK 4	4	400	00000000
COMMON BLOCK 5	5	500	00000000
COMMON BLOCK 6	6	600	00000000
COMMON BLOCK 7	7	700	00000000
COMMON BLOCK 8	8	800	00000000
COMMON BLOCK 9	9	900	00000000
COMMON BLOCK 10	10	1000	00000000
COMMON BLOCK 11	11	1100	00000000
COMMON BLOCK 12	12	1200	00000000
COMMON BLOCK 13	13	1300	00000000
COMMON BLOCK 14	14	1400	00000000
COMMON BLOCK 15	15	1500	00000000
COMMON BLOCK 16	16	1600	00000000
COMMON BLOCK 17	17	1700	00000000
COMMON BLOCK 18	18	1800	00000000
COMMON BLOCK 19	19	1900	00000000
COMMON BLOCK 20	20	2000	00000000
COMMON BLOCK 21	21	2100	00000000
COMMON BLOCK 22	22	2200	00000000
COMMON BLOCK 23	23	2300	00000000
COMMON BLOCK 24	24	2400	00000000
COMMON BLOCK 25	25	2500	00000000
COMMON BLOCK 26	26	2600	00000000
COMMON BLOCK 27	27	2700	00000000
COMMON BLOCK 28	28	2800	00000000
COMMON BLOCK 29	29	2900	00000000
COMMON BLOCK 30	30	3000	00000000
COMMON BLOCK 31	31	3100	00000000
COMMON BLOCK 32	32	3200	00000000
COMMON BLOCK 33	33	3300	00000000
COMMON BLOCK 34	34	3400	00000000
COMMON BLOCK 35	35	3500	00000000
COMMON BLOCK 36	36	3600	00000000
COMMON BLOCK 37	37	3700	00000000
COMMON BLOCK 38	38	3800	00000000
COMMON BLOCK 39	39	3900	00000000
COMMON BLOCK 40	40	4000	00000000
COMMON BLOCK 41	41	4100	00000000
COMMON BLOCK 42	42	4200	00000000
COMMON BLOCK 43	43	4300	00000000
COMMON BLOCK 44	44	4400	00000000
COMMON BLOCK 45	45	4500	00000000
COMMON BLOCK 46	46	4600	00000000
COMMON BLOCK 47	47	4700	00000000
COMMON BLOCK 48	48	4800	00000000
COMMON BLOCK 49	49	4900	00000000
COMMON BLOCK 50	50	5000	00000000
COMMON BLOCK 51	51	5100	00000000
COMMON BLOCK 52	52	5200	00000000
COMMON BLOCK 53	53	5300	00000000
COMMON BLOCK 54	54	5400	00000000
COMMON BLOCK 55	55	5500	00000000
COMMON BLOCK 56	56	5600	00000000
COMMON BLOCK 57	57	5700	00000000
COMMON BLOCK 58	58	5800	00000000
COMMON BLOCK 59	59	5900	00000000
COMMON BLOCK 60	60	6000	00000000
COMMON BLOCK 61	61	6100	00000000
COMMON BLOCK 62	62	6200	00000000
COMMON BLOCK 63	63	6300	00000000
COMMON BLOCK 64	64	6400	00000000
COMMON BLOCK 65	65	6500	00000000
COMMON BLOCK 66	66	6600	00000000
COMMON BLOCK 67	67	6700	00000000
COMMON BLOCK 68	68	6800	00000000
COMMON BLOCK 69	69	6900	00000000
COMMON BLOCK 70	70	7000	00000000
COMMON BLOCK 71	71	7100	00000000
COMMON BLOCK 72	72	7200	00000000
COMMON BLOCK 73	73	7300	00000000
COMMON BLOCK 74	74	7400	00000000
COMMON BLOCK 75	75	7500	00000000
COMMON BLOCK 76	76	7600	00000000
COMMON BLOCK 77	77	7700	00000000
COMMON BLOCK 78	78	7800	00000000
COMMON BLOCK 79	79	7900	00000000
COMMON BLOCK 80	80	8000	00000000
COMMON BLOCK 81	81	8100	00000000

VAR.	NAME	TYPE	REL.	ADDR.	VAR.	NAME	TYPE	REL.	ADDR.	VAR.	NAME	TYPE	REL.	ADDR.
TXI	RUF	101		000000	TXI	TRN	104		0000F4	TXI	TRN	104		00007C

SOURCE STATEMENT | ARFIS

TABLE	TSN	ADDR	TABLE	TSN	ADDR	TABLE	TSN	ADDR
TABLE	TSN	ADDR	TABLE	TSN	ADDR	TABLE	TSN	ADDR

COMPTON GENERATED LABELS

[illegible]

B-25

244K BYTES OF CORE NOT USED

APPENDIX C
JOB CONTROL SOFTWARE

FILE: SGMAP EXEC 5 LAPS / PHOENIX UNIVERSITY

ACONTROL OFF
SGMAP EXEC

HISTORY

MARY TOMPKINS LFNCO 11/24/80 ORIGINAL CODE

METHOD

THIS EXEC IS USED TO EXECUTE THE SGMAP PROGRAM. THE ACCEPTABLE ARGUMENTS TO THE EXEC ARE:

ARGUMENT #	DATA ON DISK	DATA ON TAPE
K1	SYMBOL FILE NAME	SYMBOL FILE NAME
K2	SYMBOL FILE TYPE	SYMBOL FILE TYPE
K3	CC FILE NAME	CC FILE NAME
K4	INPUT DATA FILENAME	TAPE NUMBER
K5	DATA FILE TYPE	FILE # ON TAPE
K6	DATA FILE MODE	TAPE DENSITY (OPTIONAL)

BOTH CC FILE AND SYMBOL FILE ARE REQUIRED TO BE ON THE USERS AISK.

EXCEPTIONS

ERROR MESSAGES FOLLOWED BY PROGRAM TERMINATION ARE FOR

1. NO TEMPORARY DISK AVAILABLE.
2. WRONG PARAMETER COUNT.
3. NON-NUMERIC TAPE NAME.
4. UNSUCCESSFUL TAPEOUNT.
5. UNEXPECTED INPUT.

PROCEDURE

```

TAG DEV PRINTER HOUSTON
SPOOL PRINTER CONT HOLD TO RSCS
GLOBAL TXLLH CMLSH FORTMOD2
CP QUERY VIRTUAL 192
&IF &RETCODE NE 0 GOTO 2M CLEAR
CP QUERY VIRTUAL 192
&IF &RETCODE NE 0 &TYPE NO TEMP DISK AVAILABLE.
&IF &RETCODE NE 0 &EXIT 1

```

```

&SPACE 3
&TYPE SYMBOL FILE: K1 K2 A
&TYPE CC FILE: K3 CC A
&TYPE INPUT DATA: K4 K5 K6
&TAPEEN = 1000

```

CHECK FOR ACCEPTABLE PARAMETER COUNT.

```

&IF &INDEX EQ 5 &GOTO -TAPE
&IF &INDEX EQ 6 &GOTO -CONT
&TYPE TOO MANY - TOO FEW INPUTS
&EXIT 2

```

```

-CONT &IF &K6 EQ 1000 &GOTO -TAPE
&IF &K6 EQ 1000 &GOTO -TAPE
&TYPE = &DATA TYPE K6
&IF &TEST EQ CHAR &GOTO -DISC
&TYPE K6 NOT THE INPUT EXPECTED.
&EXIT 3

```

```

DATA IS ON TAPE - TEST &IF TAPE NAME IS NUMERIC
&IF &TEST = &DATA TYPE K4
&IF &TEST EQ NUM &SKIP 2
&TYPE ILLFUAL TAPE NUMBER K4

```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SGMAP EXEC H LADS / PLDPR UNIVE NITY

```

      &EXIT 4
      IF &6 <> NULL THEN SET TAPDEF = &6
      &IF &INDEX EQ 6 &TAPDEF = &6
      IF TAPE HAS BEEN USED IN A PREVIOUS RUN A DISK FILE WILL
      EXIST. DO NOT MOUNT AND TRANSFER DATA.
      &DN = &CONCAT &4 &5
      STATE &DN DATA 0
      &IF &RETCOND NE 0 &SKIP 4
      &4 = &DN
      &5 = DATA
      &6 = 0
      &GOTO -DISC

      IF NO DISC FILE EXIST THEN MOUNT TAPE - TRANSFER TAPE FILE TO
      DISK.
      TAPMOUNT &4 TAP1 0 &TAPDEF
      &IF &RETCOND NE 0 &TYPE TAPMOUNT NOT MADE
      &IF &RETCOND NE 0 &EXIT 5
      &1 = &5
      TAPE REM (TAP1)
      &IF &1 EQ 0 &SKIP 1
      TAPE FOF &1 (TAP1)
      FILEDEF INMOVE TAP1 (BLKSIZE 3060 RECFM U PERM DEN &TAPDEF)
      FILEDEF OUTMOVE DISK &DN DATA 0 (LRECL 3060 BLKSIZE 3060 RECFM U PERM
      MOVEFILE
      DET 141
      &4 = &DN
      &5 = DATA
      &6 = 0

      GENERATE FILEDEFS:
      UNIT      FILE DEFINITION
      6          OUTPUT FROM CC FILE SYMBOL TABLE
      11         INPUT DATA
      12         INPUT CC FILE
      13         INPUT SYMBOL TABLE
      14         SAP OUTPUT
      30         REFAD UNIT

      -DISC &CONTINUE
      FILEDEF F111F001 DISK &4 &5 &6 (LRECL 3060 BLKSIZE 3060 RECFM U PERM
      FILEDEF F113F001 DISK &1 &2 &3 (LRECL 80 BLKSIZE 80 PERM
      FILEDEF F112F001 DISK &7 &8 &9 (LRECL 80 BLKSIZE 80 PERM
      FILEDEF F114F001 DISK OUT CC 0 (LRECL 133 BLKSIZE 133 PERM
      FILEDEF F114F001 DISK OUT DATA 0 (LRECL 133 BLKSIZE 133 PERM
      FILEDEF F114F001 DISK FILE F130F001 0 (LRECL 80 BLKSIZE 80 PERM

      BEGIN EXECUTION
      LOAD SGMAP (CLEAR NOMAP START
      EXECUTION COMPLETE -- WRAP-UP
      PRINT OUT CC 0 (CC
      PRINT OUT DATA 0 (CC
      SPOOL PRINTFW CLOSE
      &EXIT
      &END
  
```

APPENDIX D
SAMPLE OUTPUT

ORIGINAL PAGE IS
OF POOR QUALITY

INPUT SUMMARY

AT MARY TOMPKINS
DATE APRIL 22, 1981
SEGMENT H80
FILE TAPE 5450 FILE 1.NUMSY DATA.DATNUM CC
RECSKP 0
OUTPUT NUMERIC
TYPE GROUND TRUTH
*END
THIS IS AN EXAMPLE OF INPUT DATA IN SURPIXEL STRUCTURE OUTPUT AS A
NUMERICAL DUMP IN PIXEL STRUCTURE.

ORIGINAL PAGE IS
OF POOR QUALITY

JOB INITIATED ON 05/19/81 AT 11:19:26
PROGRAM SGMAC
A GROUND TRUTH MAP

CODE TRANSFORMATION
BEGIN END SYMBOL
0 0 N

PART 1 OF THE SEGMENT MAP
PIXEL MAP OF SEGMENT HH2.

ORIGINAL PAGE IS
OF POOR QUALITY

D-3

[illegible]

D-6

[illegible]

D-7

[illegible]

[illegible]

[illegible]

ORIGINAL PAGE IS
OF POOR QUALITY

D-10

D-11

[illegible]

D-12

91	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100						
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100						
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100						
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100						
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100						
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79																											

[illegible]

ORIGINAL PAGE IS
OF POOR QUALITY

[illegible]

PART 2 OF THE SEGMENT MAP.
PIXEL MAP OF SEGMENT MAP.

[illegible]

[illegible]

[illegible]

D-18

[illegible]

14

D-19

[illegible]

~~D-20~~

[illegible]

[illegible]

[illegible]

[illegible]

D-24

D-24

[illegible]

D-26

JOB COMPLETED ON 05/19/81 AT 11:20:20

ORIGINAL PAGE IS
OF POOR QUALITY

INPUT SUMMARY

AI MARY TOMPKINS
DATE APR 11 22 1941
SEGMENT
FILE CLACORN DATA.DATC1A CC.MHD CLASS
RECSKP 0
OUTPUT MAP
TYPE CLASSIFICATION
*E-10

THIS IS A PIXEL TYPE CLASSIFICATION FILE FROM CLASYFT.

ORIGINAL PAGE IS
OF POOR QUALITY

JOB INITIATED ON 05/19/41 AT 14:55:24
PROGRAM SGMAP
A CLASS/CLUSTER MAP

CONF TRANSFORMATION

HEGIN	END	SYMBOL
239	256	C
0	239	n

PART 1 OF THE SEGMENT MAP
PIXEL MAP OF SEGMENT MAP.

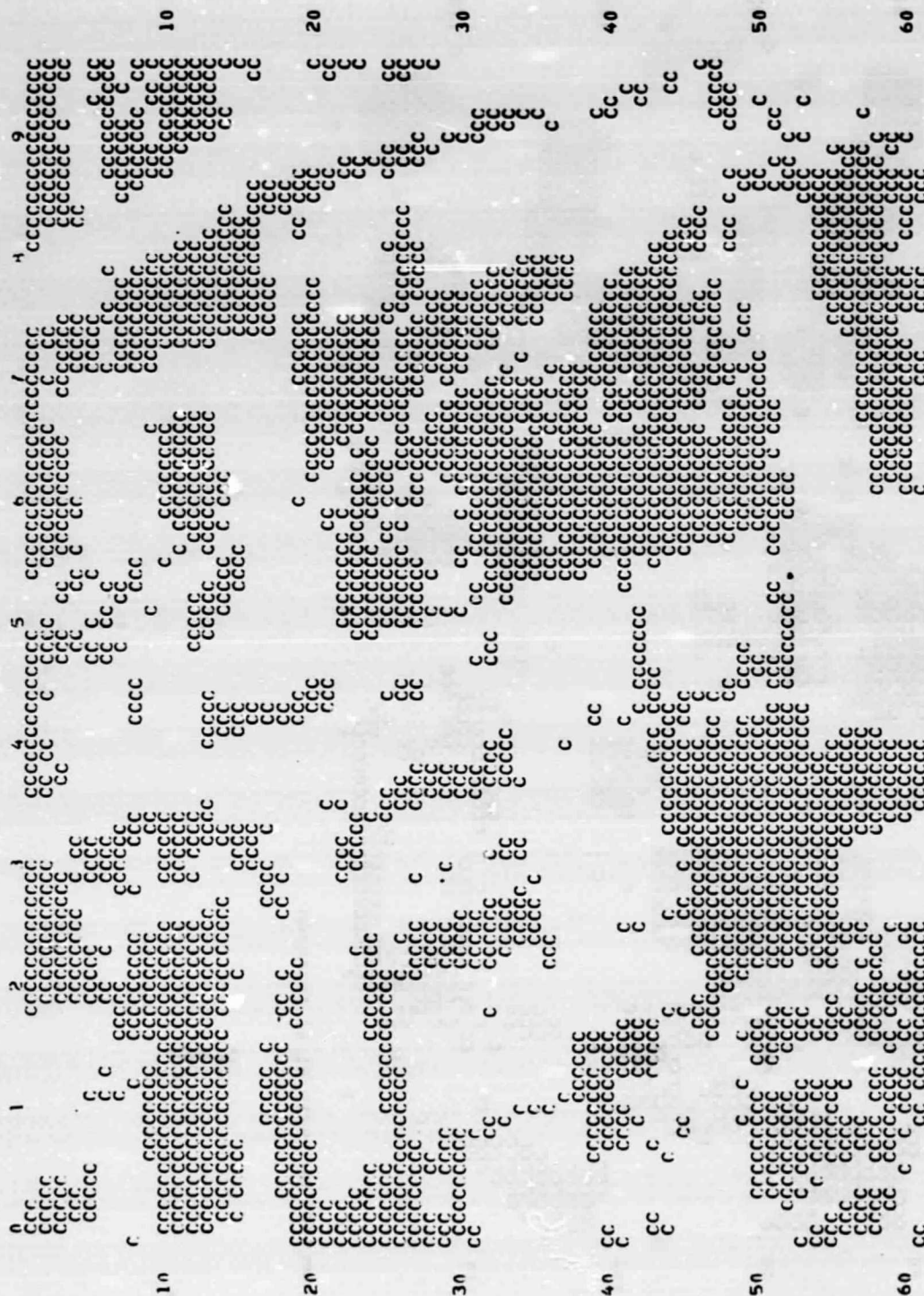


ORIGINAL PAGE IS
OF POOR QUALITY

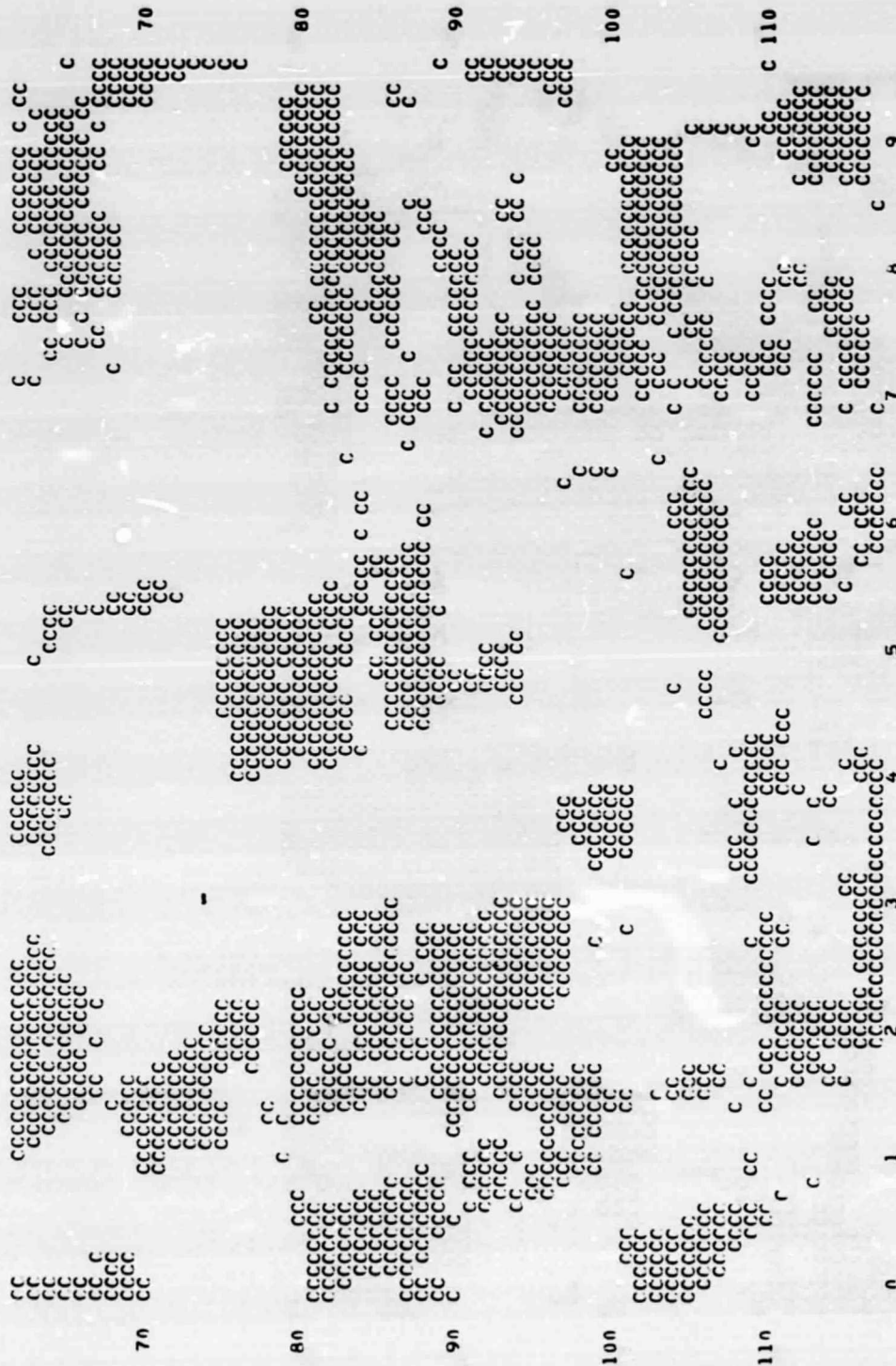


ORIGINAL PAGE IS
OF POOR QUALITY

PART 2 OF THE SEGMENT MAP
PIXEL MAP OF SEGMENT H40.



ORIGINAL PAGE IS
OF POOR QUALITY



JOB COMPLETED ON 05/19/81 AT 14:55:40

MASA-JSC